Test Plan

Our Product :

Sauce Demo From (Swag Labs)

Prepared by:

- ➔ Mohamed Magdy Moustafa Kabary
- ➔ Abdelrahman Ashraf Mohamed Abdelnaby
- ➔ Ahmed Ayman Mahmoud Mohamed
- ➔ Shahd Magdi Esmail Mohammed
- ➔ Bahaa Eldin Essam Mohamed
- ➔ Mennatu Allah Mamdouh Sedeek Osman

21.3.2025

TABLE OF CONTENTS

Page 1-> Ahmed Ayman

1.1 INTRODUCTION

This document outlines the test plan for the Sauce Demo From (Swag Labs). The purpose of testing is to ensure the website is fully functional, secure, and provides a seamless user experience. This plan covers functional, non-functional, security, and performance testing.

2.0 OBJECTIVES AND TASKS

2.1 Objectives:

The purpose of this test:

- Finding defects and ensuring that all requirements are met
- Improve product quality
- Reduce risk

2.2 Tasks:

1. **Preparing the environment for testing:**
   - Preparing an environment that combines all the requirements for the jobs
   - Preparing and preparing test data such as (user accounts, payment methods)

2. **Implementation of job tests:**
   - User registration test
   - Search test

- Marketing cart test

3. **Perform performance testing:**
    - Load testing: provides a large number of users to ensure that the site is working properly.
    - Stress testing: of the site to ensure that the site can handle visitors entering

4. **Service security tests:**
    - Authentication test: to ensure there are no vulnerabilities in the login or password recovery.
    - Penetration testing
5. **Analyze results and correct errors**
6. **Submit the final report**

3.0 SCOPE

3.1 In-Scope

- The test includes all the essential functions that ensure a successful and safe buying experience for users.
- Log in, log out and verify account creation
- Admin Panel Monitor customer reviews on the product
- Security and Performance Testing
- Ensuring that user data and sensitive information are protected

3.2 Out of Scope

- Whole egg test for servers
- Testing on browsers or devices not officially supported by the site

4.0 TESTING STRATEGY

4.1 Functional Testing:

- Verify user registration and login process.
- Check product categories, search filters, and sorting options.
- Ensure the cart updates correctly when items are added or removed.
- Validate the checkout process and payment confirmation.

4.2 UI/UX Testing:

- Verify that all elements are aligned correctly and visually appealing.
- Ensure the website is fully responsive on different devices.
- Check accessibility compliance (e.g., screen readers, color contrast).

4.3 Performance Testing:

- Load Testing: Ensure the website functions smoothly with multiple users.
- Stress Testing: Test website stability under peak traffic.

4.4 Security Testing:

- Check for SQL injection, cross-site scripting (XSS), and other vulnerabilities.
- Verify encryption of sensitive user data.
- Test session management and user authentication.

Page 2 -> Mohamed Magdy

4.1 Unit Testing

The purpose of unit testing is making sure that each component of the site SwagLabs (Sauce Demo) system operates as needed . An average of 80% coverage of statements and branches will be our goal of the testing. Error frequencies within reasonable bounds and the lack of serious defects are additional completion requirements. The test case outcomes and error logs will be used for requirements tracing.

Participants: Unit testing will be managed by the software development team.

Methodology: Automation frameworks like Selenium will be used to write and run the test scripts for unit testing.

The sequence will have:

1. Creating test cases for individual modules.
2. Running test scripts on components.
3. Logging and reviewing bug reports.
4. Iterative debugging and re-testing until completion criteria are met.

4.2 System and Integration Testing

Definition: The goal of system and integration testing is to confirm that the whole system is workable and that SwagLabs e-commerce platform modules function as a whole from all functions like :adding to cart OR payment . This includes the easy transition between functions like user authentication, checkout, and product search.

Participants: System and integration testing will be carried out by the software development team.

Methodology: Selenium will be used to write test scripts for integrated modules that concentrate on real-world situations for example : Checking all usernames that the site provides and also all the passwords and checking each of them without filling.

- To find possible problems, integration points between modules will be put through stress testing.
- For more thorough system-level testing, we may use PowerPi and similar tools may be used to analyze.
- Tests will execute and their results will be recorded in bug reports for review.

4.3 Performance and Stress Testing

Definition: SwagLabs website will be tested for performance and stress to make sure it can work within user loads and stressful situations, like an increase in users during sales events.

Resource limitations and bottlenecks will be located and addressed.

Participants: Performance and stress testing are also inside the scope of the software

Methodology:

1. Different user activity levels, from normal to high usage, will be simulated through load tests.
2. Stress scenarios will cause the system to work above its typical limits and reach its failure threshold.
3. The results will be reviewed in order to suggest and carry out performance enhancements.

Page 3 -> Mennatullah mamdouh

4.4 User Acceptance Testing (UAT)

Definition:

User Acceptance Testing (UAT) is conducted to validate that the SauceDemo website meets business requirements and is ready for operational use. The test ensures that end-users can perform essential tasks without encountering major issues.

Participants:

Name	**Role**	**Responsibility**
QA Lead	TestCoordior	Oversees the UAT process and ensures smooth execution.
Business Analyst	Requirement Validator	Confirms that the tests align with business needs.

End-Users (Selected Customers)	System Users	Execute test cases and report findings.
Developers	Issue Resolvers	Fix bugs found during UAT.

Methodology:

1. **Preparation Phase:**
    - Identify key functionalities to be tested.
    - Define acceptance criteria based on system requirements.
    - Develop UAT test scripts.
2. **Execution Phase:**
    - Users perform testing on a pre-production environment.
    - Test scenarios cover login, product browsing, cart functionality, checkout, and logout.
    - Issues are logged and categorized based on severity.
3. **Additional UAT Scenarios:**
    - **Login Tests:** Verify login with valid credentials and invalid credentials.
    - **Cart Functionality:** Add and remove multiple products from the cart.
    - **Checkout Process:** Test completing a purchase with correct and incorrect payment details.
    - **Error Handling:** Validate system behavior with incomplete or incorrect form submissions.
    - **Performance Check:** Test response time for different user interactions.
4. **Evaluation Phase:**
    - Review test results and validate against acceptance criteria.
    - Developers resolve critical issues.
    - Retest critical fixes and sign off if the system meets requirements.

4.5 Batch Testing

Definition:

Batch Testing is a process of executing multiple test cases in batches without manual intervention to ensure system reliability and efficiency. This helps in verifying the performance and stability of SauceDemo under load conditions.

Scope:

Batch Testing will cover:

- Automated login/logout operations.
- Adding and removing multiple items to/from the cart.
- Bulk checkout scenarios.
- Database and backend performance validation.

Execution Strategy:

1. **Test Case Preparation:**
   - Define automated scripts for executing batch operations.
   - Use test automation tools (e.g., Selenium, Cypress, JMeter) to run batch tests.
2. **Batch Execution:**
   - Run test cases in parallel using automation scripts.
   - Monitor system behavior and log failures.
3. **Result Analysis:**
   - Identify patterns in failures.
   - Generate batch execution reports.
   - Report critical failures for further investigation.

Expected Outcome:

- Validation of system stability under batch operations.
- Identification of performance bottlenecks.
- Assurance of smooth user experience during peak loads.

Page 3.5 -> Shahd Magdi

4.6 Automated Regression Testing

Definition:

Regression testing is the selective retesting of the SauceDemo website to ensure that modifications, such as bug fixes or new feature implementations, do not introduce unintended issues and that the website continues to function as expected.

Participants:

- QA Engineers
- Developers
- Test Automation Engineers

Methodology:

1. **Identify critical functionalities of the SauceDemo website, including:**
   - User login functionality with valid and invalid credentials
   - Adding and removing items from the shopping cart
   - Checking out and completing a purchase
   - Sorting and filtering products
   - Logging out and session handling

2. Develop automated test scripts using Selenium WebDriver with Python or Java.
3. Execute regression test suites after every major update or deployment.
4. Monitor and report any failed test cases, analyzing discrepancies with expected outcomes.
5. Maintain and update test scripts as the website evolves**.**

4.7 Beta Testing

Participants:

Selected end-users

Beta testers from within the organization

Product managers

UX/UI testers

Methodology:

Release a beta version of the SauceDemo website to selected testers.

1. Provide testers with structured test cases, including:
    - Navigating the website and reporting usability issues
    - Completing purchases and providing feedback on checkout flow
    - Testing site performance under various conditions (e.g., slow network, mobile devices)
    - Reporting any unexpected behavior or security concerns

2. Collect feedback via structured surveys and bug reports.
3. Analyze data to determine the stability and user-friendliness of the website before the official release.

5.0 HARDWARE REQUIREMENTS

Computers: Minimum requirements:

- Windows 10/macOS 11 or higher

- 8GB RAM, Intel Core i5 or equivalent
- Stable internet connection

Modems

- Minimum 10 Mbps connection for smooth website testing

6.0 ENVIRONMENT REQUIREMENTS

6.1 Main Frame

The testing environment for SauceDemo should meet the following specifications:

- **Hardware Requirements:**
  - Servers should support high concurrency for stress testing.
  - Load balancing should be enabled to test scalability.

- **Software Requirements:**
  - Operating Systems: Windows, macOS, and Linux.
  - Browsers: Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari (latest versions).
  - Automation Tools: Selenium WebDriver, TestNG/JUnit, Cypress (for frontend testing).
  - Backend Testing Tools: Postman for API testing, MySQL/PostgreSQL for database verification.

- **Mode of Usage:**
  - Standalone testing on local machines.
  - Cloud-based testing using services like BrowserStack or Sauce Labs.

- **Security Requirements:**

- Secure access to the test environment with restricted user permissions.
- Data protection measures to prevent unauthorized access to test accounts.

- **Other Testing Needs:**
  - Detailed test case documentation.
  - Access to issue tracking tools like Jira or Trello.
  - Dedicated testing workspace with proper infrastructure.

By ensuring the above environment, SauceDemo can be efficiently tested for usability, functionality, and performance

Page 4 -> Abdelrahman Ashraf

6.2 Workstation

7.0 TEST SCHEDULE

Include test milestones identified in the Software Project Schedule as well as all item transmittal events.

Define any additional test milestones needed. Estimate the time required to do each testing task. Specify the schedule for each testing task and test milestone. For each testing resource (that is, facilities, tools, and staff), specify its periods of use.

8.0 CONTROL PROCEDURES

Problem Reporting

Document the procedures to follow when an incident is encountered during the testing process. If a standard form is going to be used, attach a blank copy as an "Appendix" to the Test Plan. In the event you are using an automated incident logging system, write those procedures in this section.

Change Requests

Document the process of modifications to the software. Identify who will sign off on the changes and what would be the criteria for including the changes to the current product. If the changes will affect existing programs, these modules need to be identified.

9.0 FEATURES TO BE TESTED

Identify all software features and combinations of software features that will be tested.

10.0 FEATURES NOT TO BE TESTED

Identify all features and significant combinations of features which will not be tested and the reasons.

Page 5 -> Bahaa

11.0 RESOURCES/ROLES & RESPONSIBILITIES

No individual roles, according to Eng/ Abdul Rahman Aql.*

In this project, a collaborative approach is adopted, wherein all team members are collectively responsible for all testing activities. This strategy ensures comprehensive coverage and leverages the diverse expertise within the team. The shared responsibilities include:

- **Test Planning**: Developing and reviewing the test plan to ensure alignment with project objectives.
- **Test Design**: Creating detailed test cases and scenarios based on project requirements.
- **Test Execution**: Performing tests, recording results, and identifying defects.
- **Defect Reporting and Tracking**: Documenting identified defects and monitoring their resolution.
- **Test Reporting**: Compiling test results and metrics to assess the quality of the product.

While all team members share these responsibilities, specific tasks may be assigned based on individual expertise and project needs. This collaborative approach fosters a unified effort towards achieving the project's quality goals.

By clearly articulating this shared responsibility model, the test plan ensures that all team members understand their roles and are committed to the collective success of the testing process.

12.0 SCHEDULES

Major Deliverables, Deadlines, and Durations

Deliverable	Start Date	End Date	Notes
Project Planning & Management	March 10, 2025	March 21, 2025	Initial planning phase
Literature Review	March 21, 2025		Research and analysis
Requirements Gathering	March 10, 2025	March 20, 2025	Define project requirements
Test design work products	March 20, 2025	April 5, 2025	Define and prioritize test cases, test charters, and coverage items. Identify required test data, specify environment requirements, and create test automation strategies, including stubs, drivers, and service virtualizations.
Implementation	April 7, 2025	April 16, 2025	Develop automated test scripts, integrate code, and conduct initial testing.
Execution and completion	April 16, 2025	May 8, 2025	Run tests, validate results, fix defects, generate completion and bug reports, and finalize system readiness.

Final Presentation, Testing & Reports	May 9, 2025	Final review and presentation

To complete sections 13.0 and 14.0 of the test plan for the Sauce Demo (Swag Labs) project, the following content can be provided:

13.0 Significantly Impacted Departments (SIDs)

This section identifies the departments or business areas that may be significantly affected by the testing process or its outcomes. In the context of the Sauce Demo project, the impacted departments may include:

- **Software Development Team**: They may need to make code adjustments based on test results.

- **Quality Assurance (QA) Team**: This team is responsible for executing tests and documenting results. The testing outcomes may influence their workload and schedules.

- **Customer Support Team**: They might need to address user issues uncovered during testing.

- **Marketing Team**: Test results and product readiness could affect their promotional campaign plans.

14.0 Dependencies

This section outlines constraints or factors that may impact the testing process, including the availability of test items, resources, and deadlines. In the Sauce Demo project, dependencies may include:

- **Availability of the Testing Environment**: The testing environment must be available and stable before testing can commence.

- **Code Readiness**: The codebase should be complete and stable to begin testing.

- **Availability of Testing Tools**: Tools required for testing, such as JIRA, JMeter, and REST Assured, should be accessible and properly configured.

- **Human Resources**: Qualified team members should be available to execute tests and analyze results.

- **Project Deadlines**: Project timelines may influence the scheduling and execution of tests.

15.0 Tools

The following tools will be utilized in the testing process:

- **Automation Tools**:
    - *Apache JMeter*: For performance and load testing.
    - *REST Assured*: For testing RESTful web services.

- **Bug Tracking Tool**: *JIRA*: For project management and issue tracking.

16.0 Risks/Assumptions

Identifying high-risk assumptions is crucial for the success of the test plan. One such risk is:

- **Team's Lack of Proficiency in Using JMeter, REST Assured, and JIRA**: If team members lack the necessary skills to effectively utilize these tools, it could lead to inefficiencies and increased errors.

    Mitigation Strategies:

    - **Training Sessions**: Provide comprehensive training to ensure all team members are proficient in using JMeter, REST Assured, and JIRA.
        - **Suggested Training Resources**:
            - *Apache JMeter*:

- "JMeter بالعربي" by Mariam Elashram.
- "Performance Testing | End to End using JMeter" by Littles Law.

  - *REST Assured*:
    - "REST Assured بالعربي" by Software Quality Academy.
    - "Complete RestAssured Course" by QAcart.
  - *JIRA*:
    - "JIRA Fundamentals" on LinkedIn Learning.
    - "Atlassian JIRA for Beginners" on Udemy.
  - *Understanding System Business Requirements*:
    - "فهم متطلبات العمل الخاصة بالسيستم المراد تطويره" by Index Academy.Jira-Last 3 Videos

- **Mentorship**: Assign experienced team members to mentor those less familiar with these tools.
- **Documentation**: Provide detailed documentation and resources to assist team members in effectively utilizing these tools.

By implementing these strategies, we aim to mitigate the risks associated with the lack of tool proficiency and ensure a smooth testing process.

17.0 APPROVALS

Specify the names and titles of all persons who must approve this plan. Provide space for the signatures and dates.

Name	**Signature**	**Date**
AHMED AYMAN		2025-03-?
SHAHED MAGDI		2025-03-?
MOHAMED MAGDY		2025-03-?
ABDELRAHMAN ASHRAF		2025-03-?
BAHAA ELDIN ESSAM		2025-03-?
MENNATULLAH MAMDOUH		2025-03-?

End.